Chapter 18. Creating Documentation

Authors: *Joseph T. Buck*
Christopher Hylands
Alan Kamas
Edward A. Lee

Other Contributors:
Phil Lapsley

18.1 Introduction

Since Ptolemy is by design an extensible system, the documentation must be also be extensible. This chapter explains the document formatting conventions, scripts, and sample documents that are distributed with Ptolemy. At this time, we use a combination of text processing systems for documentation. Currently, the main systems we use are FrameMaker[1] and HTML, though older version of Ptolemy used troff, and some TeX components. A version of this particular chapter is distributed as a sample document in `$PTOLEMY/doc/samples/documents.book`[2]. It is made using FrameMaker, and can be used as a template for generating new FrameMaker documents in the Ptolemy style.

For users who do not have access to FrameMaker, a compatible alternative document formatting system based on troff is also provided. Currently, all documentation for stars, galaxies, and demo programs is based on HTML. All shell scripts and Makefiles are supplied along with the documentation so that they can be modified if necessary..

18.2 Printing the manual

The simplest way to get a hard copy version of the manual is to have a double sided bound copy sent to you. You may order the documentation set from:

EECS/ERL Industrial Liaison Program Office, Software Distribution
205 Cory Hall
University of California at Berkeley
Berkeley, CA 94720
phone: (510) 643-6687
fax: (510) 643-6694
email: ilpsoftware@eecs.berkeley.edu

If you would like to print out your own copy of the documentation, you will need a postscript printer. All of the Ptolemy documentation is contained in a collection of postscript

1. FrameMaker is a registered trademark of the Frame Technology Corporation.
2. $PTOLEMY is an environment variable that is assumed to specify the installation directory for the Ptolemy system.

files. These files have the ".ps' suffix at the end of their file names. The files are found in the Ptolemy distribution as follows:

`$PTOLEMY/doc` contains most of the documentation. Within `$PTOLEMY/doc`, the directory `users_man` contains the Ptolemy user's manual. The directory `prog_man` contains the programmer's manual. The `bin` directory contains scripts for building and printing troff based documentation. The `headers` directory contains troff header files that are needed by the scripts in the `bin` directory. The `main` directory contains the makefiles needed to print out troff based documents. Finally, the directory `$PTOLEMY/doc/samples` contains sample documents and templates to follow if you are planning to add new documentation.

Domains and their stars are documented where their source code resides. For instance, the documentation for the SDF domain is in `$PTOLEMY/src/domains/sdf/stars`. The format for this documentation, and methods for printing it are found in the section titled "Using HTML to document stars" on page 18-5.

Each of the documentation directories mentioned above may have a `README` file that explains which postscript files are which and explains how to print out the files. In any event, you can print the section you are interested in by going to the documentation directory and then printing out the postscript files found there.

18.3 Using FrameMaker

In the directory `$PTOLEMY/doc/samples`, the following files can be found:

`title.doc`:	A sample cover page.
`documents.doc`:	A sample document (very similar to this chapter).
`documentsTOC.doc`:	The table of contents for the document.
`documents.book`:	A book file that unites the above three documents.

These documents can be used as models or templates for creating new documents that are to be inserted in *The Almagest* or are to stand alone. Use `documents.doc` as a template for most applications. It defines the paragraph and character styles visible in this chapter.

By convention, except for the sample document, we do not distribute the FrameMaker files for the entire *Almagest*. Instead, we distribute the PostScript[1] code produced by Frame. The makefiles used to print manual, therefore, simply assume that the PostScript files are up to date. It is up to you to ensure this. You must also ensure that the index files corresponding to the PostScript code are up to date. The section below explains how to generate these.

18.3.1 Index Entries

We use FrameMaker to generate the indexes for each manual. Different index markers are used to denote different uses of the term bein indexed. For example, the definition of a star gets a different FrameMaker marker than a simple reference to the star. In the index file the page number of the definition will be in bold, the page number of the reference will be in a regular font.

To use the Ptolemy group markers, the following X resources should be modified in

1. PostScript is a registered trademark of Adobe Systems Inc.

the file `$FMHOME/fminit/usenglish/Maker.us`:

```
Maker*marker.10: IndexReference
Maker*marker.11: IndexExamplle
Maker*marker.12: IndexDefinition
Maker*marker.13: IndexStarRef
Maker*marker.14: IndexStarEx
Maker*marker.15: IndexStarDef
Maker*marker.24: HTMLStart
Maker*marker.25: HTMLEnd
```

These resources cause the named index markers to appear in the list of markers.

To make an index entry in a FrameMaker document, select the text you wish to appear in the index, and select the FrameMaker command Special-Marker (`Esc-s-m`). Then choose one of the above six types of index entries, using the following guidelines:

`IndexReference`:	Generic index entry
`IndexExample`:	An example of the usage of a particular feature.
`IndexDefinition`:	The definition of a term.
`IndexStarRef`:	A generic reference to a star.
`IndexStarEx`:	An example of the usage of a star. For example if the text that describes the SDF `butterfly` demo would have a index entry that looks like: `butterfly (SDF demo)`.
`IndexStarDef`:	The definition of a star. This entry is normally automatically generated when a star is compiled, so you will probably not encounter any occasion to use it directly. The text that defines the SDF `Ramp` star would have the marker text: `Ramp (SDF block)`

Avoid index entries beginning with very generic words in the Ptolemy vocabulary, like Ptolemy, `star`, `galaxy`, or `domain`. Of course, if you are writing some explanation of these basic terms, then an index entry is appropriate. Before entering index entries for a star, look in the documentation for similar stars to get an idea of the subject terms that have already been used and might be related. Be sure to follow the same capitalization rules as the existing index entries (i.e. `Ramp (SDF block)`, not `Ramp (SDF Block)`).

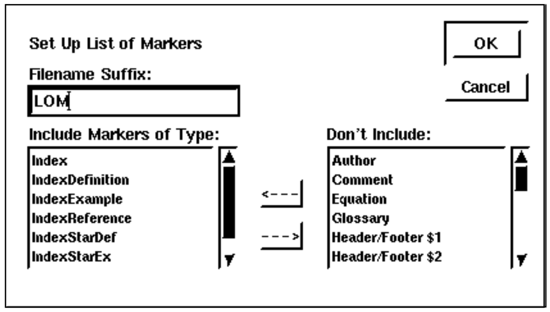
Currently we use Quadralay's WebWorks to convert Framemaker documents to html. You can use `HTMLStart` to indicate text that should be present in the html output as a link. Use a `HTMLEnd` on the end of the text of the text that represents the link. The text of the first `HTMLStart` contains the filename or URL:

```
http://ptolemy.eecs.berkeley.edu
```

The `HTMLEnd` is placed at the end of the text you want underlined in the html version.Here is a sample link to `http://ptolemy.eecs.berkeley.edu`.

The markers are not printed when the FrameMaker document is printed. WebMaker converts the text delimited by the markers into HTML hypertext links.

When you print your document, you should generate the index file that will be used to print the overall index. To do this, select File/Generate, and within the ensuing dialog box, select List/List of Markers. In the dialog box that results from this, be sure all of the above index markers are included, and then accept the default filename suffix “LOM”. The dialog box should look like this:



When you click OK, you will get a new file with a list of markers in a format acceptable to the Ptolemy index generation software. This file should be saved in “Text only” form. By convention, we name the index file using the document name with the suffix “.index”.

18.3.2 Special fonts and displays

- By convention, Ptolemy documentation uses a special font for C++ class names. In the FrameMaker template, the corresponding format is named `Class`.

```
For a display entirely set in this font,
use the "Commands" paragraph format, as shown here.
You can use "meta-Return" to force carriage returns where
you want them without getting a new paragraph.
```

- Note that if you use the Commands paragraph format, you should be sure to change any slanted double quotes “ ” to straight quotes `"`, by typing Control-Shift-`"`. In general, commands don’t have slanted double quotes, hence the need to convert them to straight quotes.
- Star and Target parameters, such as the SDF *loopScheduler* target parameter, are always in italic, use the Emphasis font format.
- The names of stars and demos should be in the `Class` format.
- If a string is to be taken literally, it should be in the `ProgramCode` character format. An example would be that the default of the *loopScheduler* target parameter is `1`. Strings such as `YES`, `NO`, `TRUE` and `FALSE`, that are used as values to parameters should also be in the `ProgramCode` format.
- The first line of a dialog with a computer should use the `Commands` paragraph format. Each successive line should use the `CommandsCont` paragraph format. The text the computer would print should be in the `ProgramCode` character format, the text the user would type should be in the **`ProgramUser`** character format.
- If a string is only an example and to be substituted with a proper value, it should be in *`ProgramVariable`* character format.
- To include images of palettes in documents, see “Capturing a screen image” on page 2-41.

18.4 Using HTML to document stars

Stars are currently documented in HTML. The `ptlang` program processes a `.pl` file and produces `.cc`, `.h` and `.html` files in the same directory as the `.pl` file. The contents of the `htmldoc` section of the `.pl` file end up in the `.html` file. See the Programmer's Manual for details.

The Tycho on-line documentation includes a style guide in `$PTOLEMY/tycho/doc/documentation.html`.